**Objective:**  
Build a collaborative to-do list application using Next.js, Tailwind CSS, Supabase, TypeScript, TanStack Query, and Zustand. The app should include real-time updates and GitHub OAuth authentication in addition to regular accounts (email – username – password).

**Requirements:**

1. **Authentication with GitHub OAuth:**
   * Implement user authentication using Supabase and GitHub OAuth.
   * Users should be able to log in and log out using their GitHub accounts.
   * Store user information securely in Supabase.
2. **Collaborative To-Do List:**
   * Create a simple to-do list where authenticated users can add, edit, and delete tasks.
   * Implement real-time updates so that all connected users see changes (like adding or deleting tasks) immediately.
   * Use Supabase's real-time capabilities to achieve this.

\*(Note)\*: all users have access to other users’ tasks and can edit them and updates should be shown in Realtime for all users.

1. **State Management:**
   * Use Zustand for managing the application's global state.
   * Ensure that the state is properly synchronized with Supabase and TanStack Query for real-time updates.
2. **Data Fetching & Caching:**
   * Use TanStack Query to fetch and cache tasks from Supabase.
   * Implement optimistic updates so that the UI reflects changes immediately while ensuring consistency with the backend.
3. **UI/UX:**
   * Build a responsive and clean UI using Tailwind CSS.
   * Provide feedback to users when actions are performed, such as adding or deleting a task.
   * Ensure that the application is accessible and follows best practices for web development.
4. **Optional Bonus Feature:**

**Task Sharing & Collaboration**:

* + Implement role-based access where users can be assigned different permissions (e.g., view-only, editor).
  + Implement a feature that allows users to assign tasks to each other.
  + Add notifications with service workers and Javascript API you can find it [here](https://developer.mozilla.org/en-US/docs/Web/API/Notification)  for users when a task is assigned to them in real-time.

**Deliverables:**

* A GitHub repository with the full source code.
* A README file explaining the setup and how to run the application locally.
* A live deployment link (e.g., on Vercel) where the task can be tested.

Username: ezeatsalex@gmail.com

Password: Ezeat$12345